BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **19**, No 2
Sofia • 2019 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2019-0012

A Review of Hashing based Image Copy Detection Techniques

Mayank Srivastava[1], *Jamshed Siddiqui*[2], *Mohammad Athar Ali*[3]

[1]*Department of Computer Engineering & Applications, GLA University, Mathura, U.P., India*
[2]*Department of Computer Science, Aligarh Muslim University, Aligarh, U.P., India*
[3]*Department of Applied Computing, University of Buckingham, Buckingham, UK*
E-mails: mayank.srivastava@gla.ac.in jamshed_faiza@rediffmail.com athar.ali@buckingham.ac.uk

***Abstract*:** *Images are considered to be natural carriers of information, and a large number of images are created, exchanged and are made available online. Apart from creating new images, the availability of number of duplicate copies of images is a critical problem. Hashing based image copy detection techniques are a promising alternative to address this problem. In this approach, a hash is constructed by using a set of unique features extracted from the image for identification. This article provides a comprehensive review of the state-of-the-art image hashing techniques. The reviewed techniques are categorized by the mechanism used and compared across a set of functional & performance parameters. The article finally highlights the current issues faced by such systems and possible future directions to motivate further research work.*

***Keywords*:** *Image forensics, Digital watermarking, Image copy detection, Hashing based image copy detection.*

1. Introduction

The unauthorized generation of duplicate multimedia content has always been far ahead of sophisticated copy detection techniques [1, 2]. Various easy to use image processing software's can do manipulations in the original images [3] and consequently it is very common to find the number of unauthorized duplicate copies of an original image. These duplicated images may have the same visual content as the original image, but their digital representations may be different. Due to this easy to copy nature of images, it is important to protect the copyright of an image [4] and therefore the identification of duplicate copies of an image is an important issue of digital rights management [5]. In many circumstances, it becomes imperative that the authenticity of the image must be verified before it is accepted. Image authentication techniques are used to prove that a received image is original and is not a duplicate copy [6]. This put us in the area of Image forensics [7], which uses a number of techniques not only to verify the authenticity of an image but also to detect unauthorized copies.

One of the most commonly used approaches in Image forensics is digital watermarking. In general, watermarking can be considered as the introduction of a signature, within an image before it is distributed [4] and its ownership can then be determined by verifying the signature [8]. A significant drawback of this scheme is that applying one or more image processing operations can easily distort or even destroy the watermark. Secondly, certain applications do not permit the embedding of a watermark since it usually leads to irreversible changes within the image, which may not be perceptual. In addition, watermark-based systems are rendered ineffective if the original image is disseminated before embedding the watermark, which might be the case in most situations. In such a situation, we require techniques that can address the drawbacks of watermarking techniques [9].

Content-based image copy detection is an image forensic technique that has gathered attention from researchers as an alternative to digital watermarking [10, 11]. Unlike watermarking, content-based copy detection techniques do not depend on embedding any mark within the image instead, the characteristics of the multimedia content itself can be used to identify its ownership [12]. The basic idea of image copy detection lies in extracting a unique image-based feature, which can be used to represent the entire image [13]. To verify the authenticity of the received image, the same set of features are extracted from both the original and the received image and compared [14]. Copy detection techniques are usually designed to be robust against image processing attacks and hence are preferred over other techniques used for image ownership and identification [15]. The applications of copy detection include authentication, usage tracking, copyright violation enforcement, etc. As an example, consider the three images shown in Fig. 1. The image shown in Fig. 1b is a copy of the image shown in Fig. 1a as it has been obtained by applying an image processing function. However, the image shown in Fig. 1c is a different image and is not a copy of either Fig. 1a or b, though perceptually similar [16]. For this example, a perfect image copy detection mechanism should detect image 1b as a copy of image 1a and treat image 1c as a different image [17]. More recently, researchers have begun to focus on hashing-based image copy detection techniques, which is the next logical step after content-based image copy detection [18, 19]. This paper, therefore, provides a comprehensive review of the state-of-the-art hashing-based image copy detection techniques. The reviewed techniques are compared by using different performance parameters to provide an overview of best-performing hashing based copy detection technique.

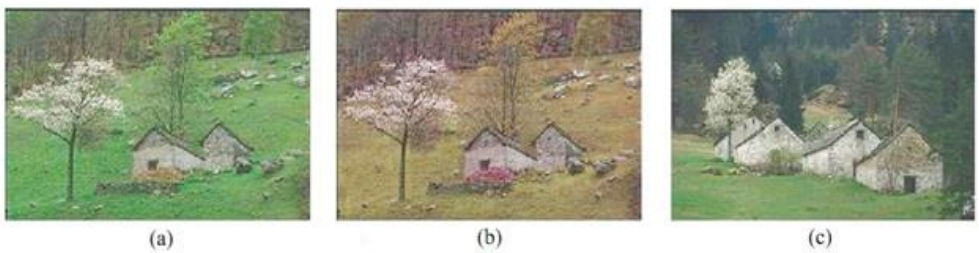

Fig. 1. Original image (a); Hue changed of image (b); Different image (c)

Rest of the paper is organized as follows: Section 2 gives the general scenario of hashing-based image copy detection system. Section 3 reviews various hashing-based image copy detection algorithms. Section 4 gives the various functional blocks used within the reviewed algorithms. Performance evaluation parameters are discussed in Section 5. Section 6 presents a comparison of the state-of-art techniques. Section 7 finally presents discussion and conclusions.

2. Hashing based image copy detection

Hashing-based image copy detection techniques are based on the underlying principle of content-based image copy detection. However, features are here compressed to form a short binary code termed as an image hash. Researchers used different ways to convert a feature to its corresponding image hash namely quantization, binary representation, vector distance etc. This image hash serves as a signature for the original image [20]. After extracting a similar signature from a test image, the signature of both original and the test image is compared. A copy detection system is expected to find a set of matching images for a given original image when the difference between their hash values is lesser than a pre-defined threshold [21, 22].

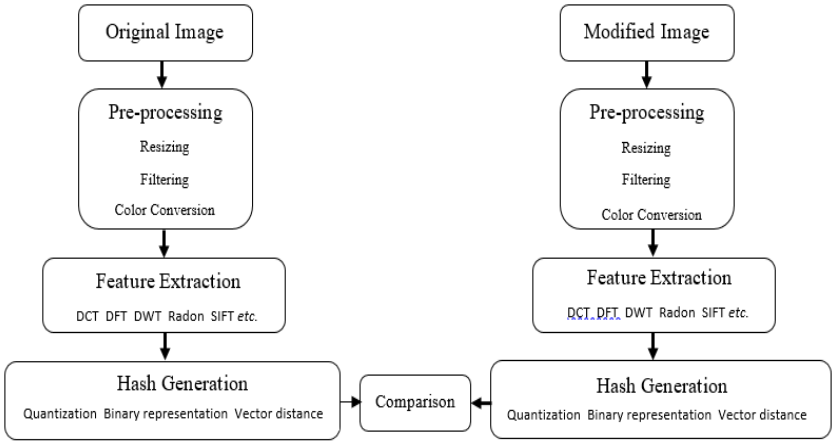

Fig. 2. Hashing based image copy detection system

However, if the threshold is too low, then a copy that has undergone minor changes will likely be skipped as being different. On the other hand, if the threshold is too high, then images which are similar but not copies will also be falsely matched and categorized as copies. Therefore, an optimal threshold value needs to be found such that a hashing mechanism results in a copy detection system that has a high true positive rate and a low false positive rate [23]. A block diagram illustrating the working of a typical hashing-based image copy detection is shown in Fig. 2.

The original image is here firstly pre-processed under operations such as resizing, filtering and color space conversion. Next, the unique features of the image are extracted by using any of the feature extraction mechanisms like DCT. Lastly,

extracted features are compressed to form a binary hash, H. The same feature extraction mechanism is then applied to the modified image to generate its hash H'. H and H' are then compared using a similarity metric, and a threshold value is used to define the outcome of a similarity metric [20]. To make the hash secure, the generated hash can be encrypted using a secure key. This ensures that the hash is key-dependent and can only be used by an authorized user for verification [24]. In general, an ideal image hash function should satisfy two basic properties – robustness and discriminability [18]. In robustness, visually identical images should produce similar hashes irrespective of their digital representation. For discriminability, different images must return a significantly different hash. Generally, these two properties conflict with each other. Therefore, to balance these properties, an optimum threshold value must be choosen so that the system can differentiate between images that are copies and those that are different [25].

3. Categorization of hashing-based image copy detection techniques

A good number of articles have been published proposing hashing-based image copy detection techniques. These techniques vary in terms of the key methodology employed as well as the overall mechanism. To provide a comprehensive review of these techniques, it is helpful to classify them into different categories depending upon the mechanism used for extracting the image features [26]. The techniques are grouped by the basic transformations used along with a few other categories (Table 1). It is worth pointing out that a few of the reviewed techniques employ more than one mechanism. In such cases, the techniques were placed in the category that it used the most.

Table 1. Categories of reviewed hashing based image copy detection algorithms

Category	Used in research papers	Brief description
DCT	[18, 27-30]	Performs well in classification and in detecting duplicate copies
DFT	[31-33]	Resistant to content-preserving alterations while preserving low collision probability
DWT	[34-42]	Robustness against non-malicious distortions such as low-pass and high-pass filtering
Radon	[23, 43-47]	Robust against rotation, scaling and translation attacks but are weak against geometric transformations
SIFT	[48-50]	Resistant to rotation [44], but exhibit a large size vector
Other	[20, 45, 51-54]	Robust against large angle rotations. Wave-atom approaches outperform DCT and DWT techniques
Block-based	[19, 21, 55]	Robust against common image manipulations like JPEG compression
Dimensionality-reduction	[56-61]	Techniques are used to reduce the size of feature matrices
Feature-based	[62-65]	Giving more emphasis to local features over global features.
Moment-based	[25, 66-68]	Used widely in classification, image matching and authentication systems
Ring-based	[24, 69-71]	Performs well specially against rotation
Other algorithms	[22, 72-82]	Rest of the image hashing algorithms.

Discrete Cosine Transform (DCT) based. Algorithms that employ DCT coefficients performed well in classification and in detecting duplicate copies [18]. Tang et al. [18] proposed image hashing based on dominant DCT coefficients for image identification. Kailasanathan, Naini and Ogunbona [27] proposed a compression-tolerant DCT image hash where the hash is constructed by considering the wavelet basis. Roover et al. [28] proposed hashing in which radial projections of the pixel luminance values based on DCT coefficients are used for hash generation. Results indicate that RASH vectors are specific to a particular image. Tang, Wan and Zhang [29] proposed a technique based on a dictionary and Non-negative Matrix Factorization (NMF) to generate image hash for identification. The proposed mechanism has a low collision probability. Longjiang et al. used a robust approach based on sign extraction of the DCT coefficients matrix of each rectangle to generate the final hash. The advantage of using DCT for feature generation is its simplicity and reduction of the DCT feature vector. However, DCT-based techniques fail against geometric transformations [26].

Discrete Fourier Transform (DFT) based. The main advantage of using DFT-based techniques is their resilience to content-preserving modifications like geometric and filtering distortions [32]. Qin and Chang [31] proposed a hashing technique based on DFT and non-uniform sampling. The proposed technique is robust to elementary image processing operations. Swaminathan [32] proposed an algorithm for hash generation using DFT features and controlled randomization. The hash is claimed to be secure against estimation and forgery. DFT-based techniques are resilient to various content-preserving operations while having low collision probability though the algorithms in this category are more complex than DCT-based algorithms [32]. Swaminathan, Mao and Wu [33] proposed an image hashing approach in which DFT is applied to the preprocessed image, which later quantized and compressed to get the final hash. The proposed approach is claimed to be resilient to common geometric and filtering operations.

Discrete Wavelet Transform (DWT) based. The algorithms discussed in this section have been proven to exhibit good time-frequency localization property. Ahmed, Siyal and Abbas [34] proposed a secure and robust hash-based scheme based on DWT where permutation sequence is used to obtain the final hash. Proposed algorithms offer robustness against non-malicious distortions such as low-pass and high-pass filtering [34]. Lu and Hsu [35] and Lu et al. [36] proposed hashing in which DWT is applied to the original image and Lowest frequency sub-band (LL) is selected for further hash generation. Lu and Hsu [35] extend their work given in [36] where extensive analysis of hashing is done for robustness, discrimination, error analysis and complexity. Michak and Venkatesan [38] proposed a robust image hash based on iterative geometric techniques. Mexiner and Uhl [37] extend image hashing proposed by [38] where pseudorandom number generator is used to produce the secure hash. Monga and Evans [39] used feature points to generate an image hash, which works well for by avoiding misclassification. Venkatesan et al. [40] proposed a novel indexing technique in which, each sub-band of the wavelet decomposition is randomly tiled into small rectangles to generate the image hash. Yang and Chen [41] proposed a novel image hash based on the

locations of significant wavelet coefficients. Karsh, Laskar and Aditi [42] propose image hashing based on DWT-SVD (DWT-Singular Value Decomposition) where *V* component of HSV color space is used for hash generation. In summary, image-hashing systems based on the LL sub-band wavelet coefficients are not robust to change in brightness, contrast enhancement, etc. Discarding the LL-band coefficients can improve the robustness and at times it may fail to detect tampering that involves modification of gray values [34].

Radon Transform (RT) based. The algorithms in this category are based on the Radon transform in which values of the projection signal is used to generate the image features which are robust to translation, scaling and rotation [23]. Lei, Wang and Huang [23] proposed a novel image hashing technique based on the Discrete Cosine Transform (DFT) and Radon transform. The algorithm performs well in detecting image copies with a small size of hash. Lefebvre, Cryz and Macq [43] proposed a robust approach based on the Radon transform and Principal Component Analysis (PCA). Lefebvre, Macq and Legat [44] present a compression and collision resilient algorithm based on the Radon transform named as RASH. Seo et al. [46] generated image fingerprints by using Radon transformation. Wu, Zhou and Niu [47] propose a hash algorithm based on radon and wavelet transform. The proposed mechanism shows resistance towards content changes. Ou and Rhee [88] proposed a five-step hashing technique based on the Radon transform. The mechanism claims to be robust and exhibits high discriminability while using a small hash of 240 bits. Unfortunately, Radon transform is not resistant to all the geometric transformations such as shifting, shearing, etc. [47]. However, this can be tolerated keeping in view that Radon transform works well for rotation, scaling and translation.

Scale Invariant Feature Transform (SIFT) based. SIFT-based techniques have proven to be more robust and effective compared to other techniques. However, their most significant drawback is the large size of the feature vector, which slows down the copy detection process [48]. Generating smaller SIFT features is a potential solution to such a problem. Chen and Hsieh [48] proposed an algorithm where 128-dimensional SIFT features are applied to the pre-processed image to produce the image hash. The proposed technique considerably reduces the execution time with a minor loss in accuracy as compared to the plain SIFT. Hefei Ling et al. [49] proposed a unique image hashing technique based on the fingerprint of local visual words. The technique surpasses similar mechanisms in terms of precision and efficiency. Lv and Wang [50] propose an algorithm similar to [49] where Harris detector is used to select the most stable key-points, which are less susceptible to image processing attacks after applying SIFT. To inherit the security of hash functions, a secret key is incorporated into either feature extraction or compression or both to make the hashes more unpredictable. SIFT-based hashing methods exhibit encouraging performance under rotation attacks and change in brightness but exhibit less-than-satisfactory performance against additive noise, blurring and compression [50].

Other transformations. The algorithms in this category are based on different transformations like Log-polar transformation that is robust to large angle rotation

operations. Wave-atom transform outperforms DCT and DWT based schemes in distinguishing spiteful tampering from non-spiteful tampering [52]. Gabor filters can be used to produce a robust and rotation-invariant hash. Ouyang, Coatrieux and Shu [20] propose the use of Quaternion Discrete Fourier Transform (QDFT) and Log-Polar Transform (LPT) for hashing. The proposed mechanism performs well as compared to similar techniques in terms of robustness against content-preserving operations and shows good sensitivity to non-authorized image content alterations. Malkin and Venkatesan [45] proposed image hashing approach based on Randlet transform to produce a key-based randomized digest. Randlet transform is built with structured randomness and hence gives good performance compared to the wavelet transform for image identification [45]. Li et al. [51] proposed a robust image hashing method based on random Gabor filter and dithered lattice vector quantization, where the gray code is employed to enhance the robustness of the hash function. Liu et al. [52] use image hashing where an input image is decomposed into five scale bands through wave atom transform. The mechanism performs better as compared to DCT and DWT based techniques. Liu and Xiao [53] proposed an image hashing approach based on Log-polar mapping and Contourlet transform for identification. Wang et al. [54] proposed a robust perceptual hashing technique using Gabor filters by using three reference matrices.

Block-based. The algorithms in this category are based on creating blocks within images for extracting image features. Tang et al. [19] proposed a robust hashing technique for color images, where Euclidean distance is used to generate an *n*-integer-based hash value. Tang et al. [21] also proposed block-based robust image hashing based on color-vector angles and DWT. The proposed mechanism is robust to normal digital operations including rotation up to 5^{o}. Yang, Gu and Niu [55] proposed four hashing algorithms wherein the first two algorithms are based on normalized block mean values while the next two utilize the rotation operation. The proposed mechanism works well against rotation up to 10^{o} except some basic image processing attacks. Algorithms in this category are robust against common image manipulations including watermark embedding, JPEG compression, contrast adjustment, brightness adjustment, Gaussian blur, gamma correction, scaling and small angle rotation [19].

Dimensionality-reduction based. This section includes techniques based on three different types of mechanisms, Non-negative Matrix Factorization (NMF), Singular Value Decomposition (SVD) and Fast Johnson-Lindenstrauss Transform (FJLT). Hernandez and Kurkoski [56] proposed image hashing in which after preprocessing, SVD is applied to get U S V components for hash generation. Kozat, Venkatesan and Michak [57] present two different scenarios of a generic hashing scheme known as SVD-SVD and DCT-SVD hashing. Lv and Wang [58] proposed a novel image hash algorithm based on the variation of FJLT named as RI-FJLT to improve the performance under rotation attacks. Monga and Michak [59] proposed two image hashing techniques named as NMF-NMF and NMF-NMF-SQ. The proposed techniques give outstanding security and robustness against a good number of image processing operations. Tang et al. [60] proposed a robust hashing method based on NMF, which exhibits a low collision probability. Karsh, Laskar

and Richhariya [61] proposed image hashing based on ring-based projected gradient-NMF and local features. The method is robust to content preserving operations and is capable of localizing the counterfeit area. Primarily SVD and NMF are applied to reduce the size of the feature matrices and to generate the final hash. SVD specially increases robustness against rotation and scaling. Fourier-Mellin transform is combined with FJLT to improve the performance towards rotation [58].

Feature-based. Algorithms in this category are based on structural features, visual model features, local features, and feature points. To extract structural features, a reference pattern is generated [64]. Some algorithms use Watson's visual model to extract visually complex features. Local features and feature point methods are also used to prove their relevance in copy detection. Monga and Evans [62] use an iterative feature detector to extract important geometry preserving feature points. Probabilistic quantization is further used on the derived features to enhance perceptual robustness. Roy and Su [63] proposed a robust hash mechanism for detecting image tampering. Tang et al. [64] constructed a perceptual image hash based on structural image features. The mechanism is sensitive to visually unacceptable alterations of the image and has a low collision probability. Xiaofeng et al. [65] proposed a visual model based on perceptual image hashing. This category makes an important contribution to the body of literature in that it proves the advantage of selecting local features over global features for copy detection.

Moment-based. Algorithms in this category use translation, scale, and rotation invariant Tchebichef and Zernike moments for hash generation. Moment-based techniques have proved to be robust against translation, scaling, and rotation and hence are widely used in classification, image matching, character recognition and authentication systems [25]. Tang, Dai and Zhang [25] proposed a perceptual hashing method for color images using invariant moments, which are invariant to translation, scaling and rotation and have been widely used in image classification and image matching. Chen, Yu and Feng [66] proposed a novel image hashing based on magnitudes of radial Tchebichef moments, which is resilient to image rotation. Zhao et al. [67] proposed perceptual image in which Zernike moments of Y and (Cb-Cr) color components are calculated to produce the final hash. Zhao and Wei [68] proposed image hashing based on Zernike moments where bits of all different blocks are combined to form the intermediate hash, which is finally pseudo-randomly permuted to produce the final hash.

Ring-based. The algorithms in this category are grouped with a reason that the image pixels of each ring is almost unchanged after rotation [70]. Different mechanisms such as image histogram, entropy, and NMF are used to generate the unique hash. Tang et al. [24] proposed an image hashing based on ring partition and invariant vector distance. Here, the L^* component of $L^*a^*b^*$ color model is used for hash generation. Tang et al. [69] proposed the use of multiple histograms for hashing. The normalized image is divided into different rings with equal area, and ring-based histogram features are extracted to make the hash resilient to rotation. The proposed mechanism is resilient to rotation of any arbitrary angle. Tang et al. [70] proposed another important hashing based on ring-based entropies. The proposed

technique performs well as compared to others in terms of time complexity. T a n g, X. Z h a n g and S. Z h a n g [71] proposed a perceptual robust hashing based on ring partition and NMF. Here, the *Y* component of the image is divided into seven rings to form a secondary image which is used to compute the image hash. These algorithms show good robustness against rotation and good discriminative capability apart from being robust to basic image processing attacks [71].

Other algorithms. This category contains algorithms which are based on a wide variety of mechanisms such as Locally Linear Embedding (LLE), Non-negative sparse coding, distributed compression, image histogram, quantization step analysis, compressive sensing, etc., G e r o l d and A n d r e a s [72] proposed a robust image hashing based on JPEG-2000 bit stream. H a d m i et al. [73] proposed a secure perceptual hashing method based on quantization step analysis. The proposed mechanism is resilient to content-preserving manipulations. J o h n s o n and R a m c h a n d r a n [74] proposed a dither-based secure image hashing for image identification. K a n g, L u and H s u [75] proposed a robust image hashing based on compressive image sensing. The key benefits of this mechanism include small and computationally secure image hash. K h e l i f i and J i a n g [76] proposed robust hashing in which a simple high-pass filter is applied horizontally and vertically to the input image to generate its filtered versions. L v and W a n g [77] proposed a semi-Supervised Spectral Embedding (SSE) mechanism for compressing real-valued intermediate image hashes into short robust binary image hashes. S u n, Y a n and D i n g [79] and T a n g et al. [22] proposed the use of LLE for formulating an image hash, which is used for image identification. X i a n g, K i m and H u a n g [80] proposed an image hashing algorithm based on image histogram. Z o u et al. [81] proposed an image copy detection framework which consists of both online and offline stages for hash generation based on non-negative sparse coding. T a n g et al. [82] proposed image hashing based on color vector angle and canny operator.

4. Functional blocks of hashing based copy detection system

A copy detection system will invariably include a number of functional blocks. This section provides an overview of various functional blocks used in different image hashing techniques. The different functional blocks include image pre-processing functions, dataset, similarity metrics, benchmark images, and image processing attacks. In the following subsections, the overview of each of these functional blocks is given.

4.1. Image pre-processing functions

The input images are pre-processed under operations such as resizing, filtering and color space conversion. During image resizing, the image is rescaled to a standard size to ensure that images with different sizes have the same hash length [19]. Image filtering is used to alleviate minor modification artifacts such as noise contamination, JPEG compression, etc. To extract unique color-based features, different color models are used. Table 2 gives a list of all the techniques used in different pre-

processing steps across different image hashing techniques. It is evident from Table 2 that most of the techniques used bilinear interpolation for image resizing, Gaussian low-pass filtering for image filtering and YCbCr as a color model for a hash generation.

Table 2. Image resizing models used by different hashing algorithms

Pre-processing function	Technique	Used in research paper(s)	Description
Image resizing	Bi-cubic interpolation	[21, 22, 25, 38, 69, 82]	Bi-cubic interpolation is an extension of cubic interpolation for interpolating data points on a two-dimensional regular grid
	Bilinear interpolation	[18, 19, 23, 24, 29, 31, 60, 61, 64, 66, 67, 68, 70, 71]	Bi-linear interpolation is an extension of linear interpolation for interpolating functions of two variables
	–	[20, 33, 41, 42, 45, 48, 51]	Resizing techniques not specified
Image filtering	Averaging	[20]	It is used to replace each pixel value in an image with the average value of its neighbors including itself
	Gaussian low-pass	[18, 21, 25, 29, 32, 33, 42, 51, 53, 60, 64, 66, 69-71, 79, 80, 82]	Gaussian filtering is used to blur images and remove noise & detail
	Linear high-pass	[67]	The filter passes signals with a frequency higher than a certain cutoff frequency and attenuates signals with frequencies lower than the cutoff frequency
	Non-linear	[31]	The linear filter is not a linear function of its output
Color model	CIE $L^*a^*b^*$	[22, 24]	In this model L^* is color lightness, a^* and b^* are chromaticity coordinates
	HSI	[25, 19]	The HSI color model describes a color in terms of how it is perceived by the human eye [41]. Here HSI indicates hue, saturation and luminance respectively
	HSV	[42]	In this model H is hue, S saturation and V is value
	RGB	[21, 82]	It is a basic color model which represents an image into three color components, i.e. Red, Green and Blue
	YCbCr	[18, 29, 31, 38, 45, 48, 60, 64, 66-71, 79]	In YC_bC_r color model, Y, which is luminance, is used
	YCbCr	[19, 25, 41, 61]	The YCbCr color model Y is luminance. Cb blue-difference chroma and Cr red-difference chroma

4.2. Image dataset

A wide range of image benchmarking dataset is available which can be used for performance evaluation of the hashing techniques under various attributes such as robustness, discriminability, etc. Each dataset contains images of a particular type such as textured images, aerial images, etc. Multiple datasets may be usually employed by researchers to verify the effectiveness of their techniques across a wide

array of different images. Table 3 above provides a list of such datasets used by different hashing algorithms. Some of the most popular datasets used among the reviewed technique are Ground Truth and USC-SIPI datasets.

Table 3. Summary of the dataset used by different hashing algorithms

Dataset	Used in research paper(s)	Web-link
Berkeley	[41]	**www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/**
BOWS2	[23]	**http://bow2.gipsa-lab.inpr.fr**
Caltech 101	[81]	http://www.vision.caltech.edu/Image_Datasets/Caltech10/
Caltech 256	[49]	http://www.vision.caltech.edu/Image_Datasets/Caltech256/
CEA CLIC	[49]	**http://www.irit.fr/RFIEC/CLIC/CLIC_kernel/CLIC_kernel.zip**
CIFAR-10	[81]	**https://www.cs.toronto.edu/~kriz/cifar.html**
Columbia University	[65]	http://www.cs.columbia.edu/CAVE/databases/
Corel 1000	[49]	**http://wang.ist.psu.edu/docs/related.shtml**
Corel Image	[36, 35]	**https://archive.ics.uci.edu/ml/datasets/Corel+Image+Features**
Ground Truth	[18, 19, 21, 22, 24, 29, 51, 58, 60, 61, 64, 69, 65, 70, 71]	http://www.cs.washington.edu/research/imagedatabase/groundtruth/
Image Net	[51]	**http://image-net.org/**
INRIA Copydays	[48]	**https://lear.inrialpes.fr/~jegou/data.php#copydays**
McGill Calibrated Color	[23, 80]	images. http://tabby.vision.mcgill.ca/
MNIST	[81]	http://yann.lecun.com/exdb/mnist/
NIST FERET	[47]	**http://www.nist.gov/itl/iad/ig/colorferet.cfm**
Oliva & Torralba	[79]	http://people.csail.mit.edu/torralba/code/spatialenvelope/
Photography	[51]	**http://www.stat.psu.edu/jiali/index.download.html**
TINY	[81]	http://horatio.cs.nyu.edu/mit/tiny/data/
UCID	[23, 20, 31, 51, 65, 66]	**http://homepages.lboro.ac.uk/~cogs/datasets/ucid/ucid.html**
USC-SIPI	[18, 19, 21, 22, 24, 28, 42, 43, 47, 61, 63, 65, 70, 71, 82]	http://sipi.usc.edu./database/

4.3. Similarity metrics

The similarity or distance metric is used to measure the performance of image hashing techniques by calculating the difference between the features of two images under comparison. Studies have shown that an appropriately chosen distance metric can significantly improve the feature matching performance of a copy detection system.

Table 4 gives a list of similarity metrics used in the reviewed techniques. Majority of hashing algorithms used normalized hamming distance, hamming distance and euclidean distance as its similarity metric for comparison.

Table 4. Similarity metrics used by different hashing algorithms

Similarity metric	Used in research papers
Bit error rate	[3, 4]
Correlation coefficient	[61, 76, 86]
Cross-correlation	[33]
Hamming distance	[1, 2, 16, 18, 19, 21-23, 25, 31, 39, 74, 75, 82, 91]
Histogram intersection	[42]
Hit rate	[4]
L1 norm	[32]
L2 Norm/euclidean distance	[5, 7, 20, 27, 35, 36, 49, 51, 77, 89, 90]
Mean square error	[33]
Min/Max based ratio	[45]
Normalized hamming distance	[17, 28, 37, 44, 52, 53, 57-59, 62, 63, 69, 79, 85, 87, 88]
Normalized L1 norm	[43]
Peak Signal to Noise Ratio(PSNR)	[64]

4.4. Benchmark images

This section discusses the benchmark images used for evaluating the performance of different hashing algorithms. Table 5 below gives a list of the benchmark images used by various image hashing algorithms. It can be seen that *Lena* is the most popular image used for evaluating the robustness of different algorithms, followed by *Baboon*, *Peppers*, *Airplane*, *House* [72, 79, 80]. All the reviewed techniques employ more than one benchmark image which is given in Table 5.

Table 5. Benchmark images used by different hashing algorithms

Image	Used in research paper(s)
Airplane	[18- 22, 24, 25, 29-31, 35, 36, 41, 42, 48, 55, 60, 64, 66, 68-71, 75, 76, 82]
Baboon	[18, 19, 21, 22, 24, 25, 29, 31, 32, 34-38, 40-43, 48, 53, 55, 58, 60, 64, 66, 68-71, 75, 79, 80, 82, 88]
Barbara	[37, 53, 55, 74, 80, 88]
Boat	[30, 37, 41, 52, 74, 76]
Bridge	[35, 36, 39]
Cameraman	[34]
Clinton	[39, 59]
Clock	[35, 36]
Couple	[31]
Escher	[78]
Goldhill	[31, 35, 36, 41, 57, 72, 74, 86]
House	[18, 19, 21, 22, 24, 25, 29, 42, 48, 60, 61, 64, 68-71, 82]
Lena	[18-25, 29-33, 35-38, 40-43, 46, 48, 52, 53, 55, 57-60, 64, 66, 69-72, 74, 75, 78-80, 82, 88, 91]
Peppers	[18-22, 24, 25, 29-32, 35-37, 41-43, 48, 53, 58, 60, 64, 66, 69-71, 74, 75, 80, 82, 88]
Sailboat	[20, 35, 36, 41, 61, 66, 80]
Splash	[35, 41, 61]
Tank	[31, 35, 41]
Toys	[39]
Truck	[37]

4.5. Image processing attacks

It is obvious that to test a copy detection system, a number of benchmark images are required along with a set of images which are close copies of each other. Copies can be created by applying a number of image processing attacks. The duplicate copies are basically used for evaluating the robustness of any of the proposed techniques. Table 6 below lists various image processing manipulations that have been employed within reported literature in order to generate duplicate copies. The image processing functions used extensively across the different reviewed technique are JPEG compression, Image rescaling, Gaussian low-pass filtering, Image rotation, Gaussian noise (additive), Cropping and Median filter.

Table 6. Image processing attacks used by different hashing algorithms

Attack	Used in research papers
Brightness adjustment	[18-22, 24, 25, 29, 37, 42, 47, 55, 61, 64, 68-71, 82]
Contrast adjustment	[18-22, 24, 25, 29, 34, 39, 42, 52, 61, 62, 64, 69-71, 79, 82]
cropping	[20, 23, 30, 32, 35, 38-41, 43, 45, 46, 48, 50, 54, 55, 57, 58, 62, 63, 67, 74, 76, 77, 80, 88]
Gamma correction	[18-25, 29, 32, 33, 42, 50, 61, 64-67, 69-71, 77, 82]
Gaussian low-pass filtering	[18-25, 28, 29, 31, 34, 36, 38, 39, 41-43, 46, 52, 55, 61, 62, 64, 66, 68-71, 73, 74, 79-82, 88]
Gaussian noise (additive)	[20, 23, 29, 31-33, 35, 37-39, 45, 47, 50-52, 54, 55, 58, 60, 62, 64-68, 73, 76, 77, 79, 80, 88]
Image rescaling	[18-25, 28, 29, 32, 35, 36, 38-43, 45-47, 50, 53-55, 58, 60-62, 64-71, 74, 76, 77, 79-82, 88]
Image rotation	[18, 20, 23-25, 28, 31, 32, 35, 38-40, 42, 43, 45, 46, 50, 52, 54, 57, 58, 62, 63, 65-67, 74, 76, 77, 79-81, 88]
JPEG compression	[2, 18-25, 28, 29, 31-43, 45-47, 50, 51, 53-55, 57, 60-71, 73, 76-82, 88]
Median filter	[20, 23, 32, 33, 35, 36, 38-41, 46, 47, 51-53, 55, 62, 66, 74, 76, 79-81, 88]
Salt & pepper noise	[22-24, 31, 42, 50, 52, 53, 55, 58, 65, 66, 77, 88]
Shearing	[23, 30, 32, 35, 38, 39, 41, 50, 55, 62, 76, 77, 80, 81]
Watermark embedding	[18, 19, 21, 22, 24, 25, 29, 42, 60, 61, 64, 66, 69-71, 79, 81, 82, 88]

5. Performance parameters

This section examines the various performance parameters that are used to evaluate the algorithms that have been reviewed in this article. Different performance parameters like robustness, discrimination and Receiver Operating Characteristics (ROC) are discussed in the following subsections.

5.1. Robustness

Robustness of the hashing algorithm is evaluated to determine its capability to resist various types of image preprocessing attacks. A perfect image-hashing algorithm identifies all its preprocessed variants as multiple copies of the original image. Table 7 shows the robustness parameters generated by the reviewed hashing algorithms, which are obtained by calculating the difference between the hash values of original images and its duplicate copies. It includes parameters such as *maximum*, *minimum*, *mean* and *standard deviation*. The table also includes a column named as the *next highest value* and *visual identification*. The *next highest value* corresponds to the

second highest maximum difference value between the original image and its preprocessed variant(s) while *visual identification* represents the percentage of visually identical images correctly identified as similar images. It is quite obvious that a lower mean along with a lower standard deviation implies that the difference between the hash values is small and they exist in the vicinity of the mean. Such an observation can lead to the deduction that various attacks performed on the original image have been largely ineffective and that the attacked images are very similar to the original image.

To perform a fair comparison, a normalized mean and normalized standard deviation is employed which is shown in Table 7. Normalized values are obtained by dividing the mean and standard deviation of the techniques with their threshold value. From Table 7, it is clear that [24] has the lowest normalized mean along with a low value of normalized standard deviation. This indicates that the difference between the hash values of the original image and its attacked version is small and thus we can say that technique reported in [24] is resilient to most of the image processing attacks. In contrast, the technique in [22] returns a large value of the normalized mean, which implies that the images underwent significant changes as a result of various attacks. Also, the mechanism which returns a higher percentage of true positives and a low percentage of false positives considered ideal. The visual identification parameters are calculated by selecting an appropriate threshold value. Based upon the values given for visual identification in Table 7 [24] again claims to produce the best performance.

5.2. Discriminability

Discriminability is used to measure the performance of hashing algorithms when dissimilar and visually different images are compared to the original image. [20, 23]. Theoretically, the discrimination parameter should have a higher normalized mean & higher minimum and maximum difference values as compared to the robustness parameter. Table 8 gives the discrimination values for some of the most popular hashing techniques that have been reported in the literature. It can be seen from Table 8 that all the parameters for discrimination are much higher compared to robustness for the reviewed techniques listed in Table 7.

Table 7. Robustness parameters generated by various reported techniques. Table gives average values for different processing operations: Norm. – Normalized; Thre. – Threshold; VI – Visual Identification

Paper	Max	Next highest value	Min	Mean	Std. Dev.	Thre.	Norm. Mean	Norm. Std. Mean	VI (in %)
[18]	14	6	0	2.1525	2.07889	10	0.2151	0.2078	98.5
[19]	8.14	5.46	0	1.40875	0.97571	5	0.2817	0.1951	92.58
[21]	13866	12217	0	3632.25	2064.88	10000	0.3632	0.2064	97.74
[22]	1	0.999	0.4572	0.96244	0.02842	0.7	1.3749	0.0406	99.53
[24]	260.52	215	0	33.991	27.218	200	0.1699	0.1360	99.56
[25]	12.05	7.32	0.01	3.55625	1.3425	8	0.4445	0.1678	93
[31]	0.2467	0.2184	0.026	0.1158	–	0.2	0.5790	–	–
[69]	246784	47988	13	10992.1	14006	48000	0.2290	0.2917	–
[70]	–	–	–	0.997	0.0056	0.95	1.0494	0.0058	99.43

For example, the normalized mean values for discrimination and robustness for [18, 19, 21, 24, 25, 69] are 2.78, 2.87, 2.16, 3.20, 2.80, 7.17 and 0.21, 0.28, 0.36, 0.16, 0.44, 0.22 respectively. It is evident that discrimination values are much higher

compared to robustness values, which is a good indicator of the performance of the reviewed techniques. In particular, the technique proposed in [69] exhibits the best performance. It is important here to clarify that the range of robustness values should be lower than the range for discrimination. The larger the gap between them, better is the performance. A technique exhibiting such behavior is considered to be the most promising. However, these two ranges tend to overlap due to the presence of outliers in the analyzed data.

Table 8. Discriminability exhibited by various reported techniques. Values are taken from robustness table

Paper	Max	Min	Mean	Std. Dev.	Threshold*	Normalized Mean	Normalized Std. Dev.
[18]	46	6	27.8	5.23	10	2.78000	0.52300
[19]	35.19	5.41	14.37	3.43	5	2.87400	0.68600
[21]	62460	3628	21630	9157	10000	2.16300	0.91570
[22]	0.6967	–0.6891	0.0066	0.1905	0.7	0.00943	0.27214
[24]	1329.71	204.9	640.17	154.53	200	3.20085	0.77265
[25]	51.36	6.46	22.46	7.37	8	2.80750	0.92125
[31]	–	–	0.443	0.027	–		
[69]	4041011	9920	344567	636826.05	48000	7.17848	13.26721
[70]	0.9828	–0.9625	0.2091	0.4584	0.95	0.22011	0.48253

To perform a more consistent analysis, we considered the second largest value for robustness in Table 7 (column 3). The second largest robustness values and the minimum discrimination values (Table 8) in [18, 19, 24]. [25] are 6, 5.46, 215, 7.32 and 6, 5.41, 204.9, 6.46, respectively. It is evident that these values only overlap slightly, which is acceptable. Among the reviewed techniques [18] returns the best performance as it is having 6 as its next highest value for robustness and minimum value for discrimination, which clearly shows the separation between robustness and discrimination. Similarly, the maximum (Max) difference values for robustness (Table 7) and discrimination (Table 8) for [18, 19, 21, 24, 25, 69] are 14, 8.14, 13866, 260.52, 12.05, 246784 and 46, 35.19, 62460, 1329.71, 51.36, 4041011, respectively. It can be seen that the maximum difference values of discrimination of most of the reviewed techniques are almost three times the maximum difference values of robustness except in [22]. This again serves as a good indicator for the performance of these algorithms. Based on the values given above, [69] again claims to exhibit the best performance.

5.3. Receiver Operating Characteristics (ROC)

Majority of hashing algorithms employ the ROC curve to indicate classification performance between robustness and discrimination. However, parameters used to draw the ROC curve along with the thresholds can vary [18, 48, 70, 79]. Table 9 below lists the different ROC parameters adopted by various image hashing techniques reviewed in this paper. It is important here to emphasize that different authors used different terms for the same set of parameters. From the table, it is evident that most of the reported techniques employed TPR vs FPR values to draw the ROC curve for different thresholds. Ideally, the ROC curve should pass through (0, 1) where 0 is for false positive rate and 1 is for true positive rate. Consequently [22, 19, 24, 54] exhibit the best ROC curves among the reviewed techniques.

Table 9. ROC parameters used by reviewed algorithms. n_1 is the number of different images detected as identical images; n_2 is the number of identical images detected as different images; n_3 is the number of similar images considered as visually identical images; N_1 is the total number of different images; N_2 is the total number of identical images.

ROC parameter	Research paper(s)	Formula
False Positive Rate (FPR) / False Negative Rate (FNR)	[34, 39, 45, 47, 52, 55, 59, 65-67, 79]	$\text{FAR} = \left(\frac{n_1}{N_1}\right)$&FRR=$\left(\frac{n_2}{N_2}\right)$
False Accept Rate (FAR) / False Reject Rate (FRR)		
Probability of False Positive / Probability of False Negative		
Probability of False Alarm / Probability of Miss		
Prob. of Correct Detection (TPR) /False Rejection Rate (FRR)	[51]	$\text{TPR} = \left(\frac{n_3}{N_2}\right)$&FRR=$\left(\frac{n_2}{N_2}\right)$
Recall Rate (RR)/ Precision Rate (PR)	[35, 48, 49, 81]	$\text{RR} = \left(\frac{n_3}{n_3+n_2}\right)$&PR=$\left(\frac{n_3}{n_3+n_1}\right)$
False Positive Rate (FPR) / True Positive Rate (TPR)	[18-25, 32, 33, 42, 50, 58, 61, 63, 69-71, 76, 77, 82]	$\text{FPR} = \left(\frac{n_1}{N_1}\right)$&TPR=$\left(\frac{n_3}{N_2}\right)$
Probability of False Alarm / Probability of Correct Detection		
Probability of False Alarm / Probability of True Detection		

6. Comparison with the state-of-the-art techniques

Table 10. State-of-the-art techniques used for comparison by reviewed hashing algorithms

Ref. No	Technique used	Used by research paper(s)
[48]	Used for identification of audio clips and database lookups	[73]
[72]	Color vector angles and Discrete wavelet transform	[22]
[71]	Color vector angles	[82]
[51]	Geometry preserving feature points	[54]
[34]	Random Gabor filtering and dithered lattice vector quantization	[18, 21, 22, 24, 71, 82]
[30]	Color images based on hypercomplex representation	[85]
[68]	Fourier transform and controlled randomization	[23, 42, 47, 63]
[70]	Invariant moments	[24, 71]
[49]	Iterative geometric techniques	[32, 33]
[67]	Locally linear embedding	[22]
[66]	Content feature extraction and cryptographic concepts	[73]
[73]	Multiple histogram	[18, 21]
[78]	Non-negative matrix factorization	[29, 61, 67]
[52]	Non-negative matrix factorization and weight vectors	[24, 29, 47, 50, 58, 60, 61, 64, 65, 67, 70, 71, 76]
[14]	Probabilistic quantization based on discrete wavelet transform and Radon transform	[88]
[61]	Radial projection of image pixels	[29, 31, 60, 63, 64, 71]
[81]	Ring-based entropies	[24]
[84]	Randomized signal processing	[32, 33, 42, 52, 63, 66, 76]
[83]	Ring partition and non-negative matrix factorization	[20, 42]
[82]	Ring partition and invariant vector distance	[42]
[54]	Radon transform and DCT	[18, 19, 21-24, 69, 70, 82]
[33]	Image hashing based on Radon transform and Discrete Fourier transform	[24, 71]
[53]	Salient feature points and Hausdorff distance measure	[63, 76]
[29]	Singular value decomposition	[19, 21, 25, 42, 52, 59, 63, 66, 69, 70, 71, 76]
[95]	Zernike moments and local features	[20, 42, 61, 65]

Performance evaluation and comparison with the state-of-the-art is an essential step in verifying the efficacy of a proposed technique. This comparison can be either based on all performance parameters mentioned above or on a subset of them. The performance of reviewed hashing algorithms is evaluated by comparing it with some of the state-of-the-art techniques [57, 59, 88]. Table 10 above gives a list of state-of-the-art techniques used by the reviewed techniques for comparison. It can be observed that the techniques NMF-NMF-SQ and SVD-SVD are most frequently used as a benchmark for comparative analysis.

6.1. Accuracy

To measure the performance of the hashing techniques, ROC curves are the most popularly used metric. The True Positive Rate (TPR) and the False Positive Rate (FPR) are used to draw the ROC curve where TPR and FPR correspond to robustness and discrimination respectively. The optimal TPR values when FPR is 0 and optimal FPR values when TPR is 1 are depicted in Table 11 for the reviewed hashing algorithms across various processing environments. Here, the processing environment represents the simulated environment that is used by the authors for executing their implementation along with the execution of state-of-the-art techniques for comparison. The processing environment varies depending on the processor type, size of RAM and the simulation software employed. The ideal optimal FPR and TPR values for any technique should be 0 and 1, respectively. It is evident from Table 11 that technique which has given processing environment, performs well among all compared techniques.

Table 11. Optimal TPR and FPR values generated by the reviewed techniques

Reference No	Optimal TPR when FPR=0								Optimal FPR when TPR=1							
	Processing environments given by								Processing environments given by							
Tech.	[18]	[19]	[21]	[22]	[24]	[69]	[70]	[71]	[18]	[19]	[21]	[22]	[24]	[69]	[70]	[71]
[18]	0.969								0.012							
[19]		0.98								0.007						
[21]			0.8	0.941							0.16	0.103				
[22]				0.995								0.009				
[23]					0.976			0.877					0.145			0.145
[24]					0.999								0.001			
[25]					0.959		0.945	0.93					0.031		0.031	0.031
[28]								0.91								0.269
[51]	0.623		0.53	0.783	0.725			0.517	0.448		0.91	0.704	0.911			0.911
[57]		0.4	0.4			0.11	0.733	0.126		0.82	0.82			0.95	0.823	0.945
[59]					0.932		0.91	0.813					0.142		0.142	0.535
[69]	0.676		0.7			0.767			0.797		0.74			0.62		
[70]					0.972		0.976						0.041		0.008	
[71]								0.983								0.001
[79]				0.896								0.557				
[88]	0.637	0.6	0.6	0.702	0.638	0.55	0.518		1	1	1	0.996	1	1	1	

6.2. Execution time

In this section, the execution time for generating the image hashes of the compared algorithms are given. The time taken to extract image hashes of 200 images for the discriminative analysis is noted for each of the algorithms and then the average time for generating an image hash is found. Table 12 shows different reviewed algorithms

along with their execution time (seconds) on eight different processing environments. The comparison was made by using a dataset of 200 images which consists of 67 internet-based images, 33 images taken from a digital camera and 100 images from the Ground truth database. It is important here to specify that dataset is same for all the processing environments but we obtained different execution values for the same reviewed techniques. This could be due to a combination of factors such as varying hardware configurations, different versions of simulation software and the lack of exact specifications for the images within the dataset. Different techniques excel in their execution performance in different processing environments. Techniques reported by [69, 21, 69, 79, 70, 24, 60, 70, 51] exhibit the best execution results in processing environments [18, 21, 22, 24, 42, 61, 70, 71], respectively. All the papers have represented computational complexity in terms of execution time except [47] whose computational complexity is $O(N^2 \lg N)$.

Table 12. Execution time (in seconds) of reviewed hashing algorithms

Reference No	Processing environments given by							
	[18]	[21]	[22]	[24]	[42]	[61]	[70]	[71]
[18]	0.162							
[21]		0.1	0.27					
[22]			0.61					
[23]				3.422				12.8
[24]				0.286	0.28			
[25]				0.616			1.429	2.6
[28]								9.6
[40]					2.4			
[42]					2.1			
[51]	0.285	0.67	0.42	0.903				0.67
[57]		0.28			1.5		0.65	1.5
[59]				0.926		2.98	1.153	2.4
[60]						0.93		
[61]						2.1		
[67]					2.12	2.4		
[69]	0.137	0.1						
[70]				0.098			0.437	
[71]					2.8			2.8
[76]						2.6		
[79]			0.04					
[88]	0.455	4.58	3.04	3.333			6.51	

6.3. Size of feature vector

The size of the feature vector of the resulting hash is one of the important parameters for any of the hashing system. Generally, a large hash size implies that a longer processing time is needed for comparison [24]. Also, storing large-sized hash for each of the images of the dataset will involve a significant storage overhead [49]. Therefore, it is desirable to have a small hash size to ensure optimum performance of the algorithm while also optimizing other affected parameters. Table 13 provides the hash sizes for the different algorithms reviewed in this paper.

Table 13. Hash sizes of reviewed hashing algorithms

Category	Reviewed techniques	Hash size
DCT	[18, 29, 30]	64 bits, 512 bits, 512 bits
DFT	[31, 32]	444 bits, 420 bits
DWT	[34-36, 40, 42]	896 bytes, 240 bits, 805 bits, 80 digits, 64 bits
Radon	[23, 46, 88]	150 bits, 440 bits, 240 bits
SIFT	[49, 50]	32 bits, 320 bits
Other transformations	[20, 45, 51, 52]	224 bits, 300 bits, 120 bits, 405 bits
Block-based	[19, 21]	64 bits, 960 bits
Dimensionality-reduction	[57, 59-61]	64 digits, 320 bits, 848 bits, 1600 digits
Feature-based	[63, 64]	920 bits, 896 bits
Moment-based	[25, 66-68]	42 digits, 140 bits, 560 bits, 3528 bits
Ring-based	[24, 69-71]	40 digits, 16 digits, 64 digits, 64 digits
Others	[22, 75-77, 79, 80]	550 bits, 368 bits, 250 bits, 320 bits, 300 bits, 435 bits

SIFT [49] has the smallest hash size of 32 bits; however, other performance parameters were not very encouraging. It is important to note that the table below consists of hash size in two different units, i.e., binary [18, 29, 30] and decimal digits [24, 69, 70]. Ideally, an attempt is made to realize the smallest hash size possible without compromising on various performance parameters.

6.4. Key dependence

To view the key-dependent performance of hashing algorithms, different key values are used to generate hashes keeping rest of the parameters unchanged. It is important here to specify that out of the entire set of keys used for hash generation, one is correct while the rest of the keys are incorrect. The distance between the image hash with the correct key and incorrect keys are calculated to give the outcome. Table 14 lists the various parameters related to key dependence.

Table 14. Key dependence parameters generated by reviewed hashing algorithms. * Indicates average values for different image processing attacks except for rotation. ** Indicates highest of mean values for different image processing attacks.

Parameters	Research papers No								
	[18]	[20]	[21]	[24]	[31]	[34]	[46]	[52]	[72]
Threshold	10	0.2	10000	200	0.2	48	0.1928**	0.1859*	>0.1
Minimum of difference of hash value after changing the key	>15	>0.2	18514	625	Close to 0.2	240	0.4	0.4222-0.5630	>0.2

Here, *Threshold* indicates the difference of hash values between two images, which is used in robustness analysis (Table 7), which indicates that below which images are the same. The *Min hash value after changing the key* is a parameter that is used to define the minimum difference between the hash values of an image based on the original key and number of incorrect keys. It can be seen that in most of the cases, the minimum difference between the hash values after changing the key is very large as compared to its threshold value [18, 21, 24, and 34]. This result indicates that given image hashing techniques are highly key dependent and hence it is very

difficult for an attacker to estimate the hash value without knowing the correct key. The technique in [34] exhibits the best performance as it returns the largest difference of the hash value after changing the key compared to the threshold.

7. Discussion and conclusion

This paper has presented a comprehensive review of hashing based image copy detection techniques for image identification. Firstly, the study presented a general framework which most of the copy detection techniques have adopted over the last few years. Secondly, the reviewed techniques are categorized and discussed on the basis of basic transformation employed. Lastly, different functional and performance parameters of the reviewed techniques are presented. In this review, research has proceeded along the following directions: (1) Reducing the complexity of the algorithm, which can be achieved by generating a feature vector of a smaller hash size; (2) Increasing the robustness of the algorithms, which can be achieved by using strong features that are invariant to a wide range of image processing operations; (3) Introducing the concept of key dependence to make the hash generation process key-dependent and hence more secure. The review also indicates that the techniques that employ block-based features and ring-based features returned better functional and performance parameters.

During the course of this review, some drawbacks were also noticed. Many techniques are unable to handle all types of images such as heavily textured images. Some of the algorithms are heavily dependent on various parameters and the setting of different thresholds. Most of the algorithms are not giving promising result against a specific image processing operation "rotation". The review also identified the lack of a single standardized benchmark image dataset for performance evaluation. Another hurdle that made comparative analysis difficult is the lack of a single standardized metric for calculating the similarity and the lack of a common parameter to draw the Receiver Operating Characteristics (ROC) curve. Finally, it is expected that classy and reliable copy detection algorithms will be developed by overcoming the drawbacks and challenges mentioned above.

References

1. Lian, S., D. Kanellopoulos. Recent Advances in Multimedia Information Systems Security. – Informatica – An International Journal of Computing and Informatics, Vol. **33**, 2009, pp. 3-24.
2. Qureshi, M. A., M. Deriche. A Bibliography of Pixel-Based Blind Image Forgery Detection. – Signal Processing: Image Communication, Vol. **39**, 2015, No A, pp. 46-74.
3. Liu, G., J. Wang, S. Lian, Z. Wang. A Passive Image Authentication Scheme for Detecting Region-Duplication Forgery with Rotation. – Journal of Network and Computer Applications, Vol. **34**, 2011, No 5, pp. 1557-1565.
4. Qazi, T., K. Hayat, S. U. Khan, S. A. Madani, I. A. Khan, J. Kolodziej, H. Li, W. Lin, K. C. Yow, C. Z. Xu. Survey on Blind Image Forgery Detection. – IET Image Processing, Vol. **7**, 2013, No 7, pp. 660-670.

5. K a n g, X., S. W e i. An Efficient Approach to Still Image Copy Detection Based on SVD and Block Partition for Digital Forensics. – In: IEEE International Conference on Intelligent Computing and Intelligent Systems (ICIS), 2009, pp. 457-461.
6. B a t t i a t o, S., G. M. F a r i n e l l a, E. M e s s i n a, G. P u g l i s i. Robust Image Alignment for Tampering Detection. – IEEE Transactions on Information Technology Forensics and Security, Vol. **7**, 2012, No 4, pp. 1105-1117.
7. R e d i, J. A., W. T a k t a k, J. A. D u g e l a y. Digital Image Forensics: A Booklet for Beginners. – Multimedia Tools and Applications, Vol. **51**, 2011, No 1, pp. 133-162.
8. H a n, B., J. L i. A New Zero-Watermarking Algorithm Resisting Attacks Based on Differences Hashing. – Cybernetics and Information Technologies, Vol. **16**, No 2, 2016, pp. 135-147.
9. W u, M. N., C. C. L i n, C. C. C h a n g. Novel Image Copy Detection with Rotating Tolerance. – Journal of Systems and Software, Vol. **80**, 2007, No 7, pp. 1057-1069.
10. C h a n g, E., J. Z. W a n g, C. L i, G. W i e d e r h o l d. RIME: A Replicated Image Detector for the World-Wide Web. – In: Proc. in SPIE Multimedia Storage and Archiving Systems III, Vol. **3527**, 1998.
11. H s i a o, J. H., C. S. C h e n, L. F. C h i e n, M. S. C h e n. A New Approach to Image Copy Detection Based on Extended Feature Sets. – IEEE Transactions on Image Processing, Vol. **16**, 2007, No 8, pp. 2069-2079.
12. S z u c s, G., D. P a p p. Content- Based Image Retrieval for Multiple Objects Search. – Cybernetics and Information Technologies, Vol. **17**, 2017, No 2, pp. 106-118.
13. N a s s i h, B., M. N g a d i, A. A m i n e, A. E l-A t t a r. New Proposed Fusion between DCT for Feature Extraction and NSVC for Face Classification. – Cybernetics and Information Technologies, Vol. **18**, 2018, No 2, pp. 89-97.
14. A r n i a, F., A g u s t i n u s, K. M u n a n d i, M. F u j i y o s h i, H. K i y a. Content-Based Image Copy Detection Based on Sign of Wavelet Coefficients. – In: International Workshop on Advanced Image Technology, 2011.
15. W a n g, Y. G., Y. L e i, J. H u a n g. An Image Copy Detection Scheme Based on Radon Transform. – In: IEEE 17th International Conference on Image Processing, 2010, pp. 1009-1012.
16. K i m, C. Content-Based Image Copy Detection. – Signal Processing: Image Communication, Vol. **18**, 2003, No 3, pp. 169-184.
17. S e b e, N., Y. L i u, Y. Z h u a n g, T. H u a n g, S. F. C h a n g. Blind Passive Media Forensics: Motivation and Opportunity. – In: International Conference on Multimedia Content Analysis and Mining, Vol. **4577**, 2007, pp. 57-59.
18. T a n g, Z., F. Y a n g, L. H u a n g, X. Z h a n g. Robust Image Hashing with Dominant DCT Coefficients. – Optik, Vol. **125**, 2014, No 18, pp. 5102-5107.
19. T a n g, Z., X. Z h a n g, X. D a i, J. Y a n g, T. W u. Robust Image Hash Function Using Local Color Features. – International Journal of Electronics and Communications (AEU), Vol. **67**, 2013, No 8, pp. 717-722.
20. O u y a n g, J., G. C o a t r i e u x, H. S h u. Robust Hashing or Image Authentication Using Quaternion Discrete Fourier Transform and Log-Polar Transform. – Digital Signal Processing, Vol. **41**, 2015, pp. 98-109.
21. T a n g, Z., Y. D a i, X. Z h a n g, L. H u a n g, F. Y a n g. Robust Image Hashing via Colour Vector Angles and Discrete Wavelet Transform. – IET Image Processing, Vol. **8**, 2014, No 3, pp. 142-149.
22. T a n g, Z., L. R u a n, C. Q i n, X. Z h a n g, C. Y u. Robust Image Hashing with Embedding Vector Variance of LLE. – Digital Signal Processing, Vol. **43**, 2015, pp. 17-27.
23. L e i, Y., Y. W a n g, J. H u a n g. Robust Image Hash in Radon Transform Domain for Authentication. – Signal Processing: Image Communication, Vol. **26**, 2011, No 6, pp. 280-288.
24. T a n g, Z., X. Z h a n g, X. L i, S. Z h a n g. Robust Image Hashing with Ring Partition and Invariant Vector Distance. – IEEE Transactions on Information Forensics and Security, Vol. **11**, 2016, No 1, pp. 200-214.

25. T a n g, Z., Z. Y. D a i, X. Z h a n g. Perceptual Hashing for Color Images Using Invariant Moments. – Applied Mathematics and Information Sciences, Vol. **6**, 2012, No 2S, pp. 643S-650S.
26. A l-Q e r s h i, O. M., B. E. K h o o. Passive Detection of Copy-Move Forgery in Digital Images: State-of-the-art. – Forensic Science International, Vol. **231**, 2013, No 1-3, pp. 284-295.
27. K a i l a s a n a t h a n, C., R. S. N a i n i, P. O g u n b o n a. Compression Tolerant DCT Based Image Hash. – In: Proc. of 23rd International Conference Distributed Computing Systems Workshop, 2003, pp. 562-567.
28. R o o v e r, C. D., C. D. V l e e s c h o u w e r, F. L e f e b v r e, B. M a c q. Robust Video Hashing Based on Radial Projections of Keyframes. – IEEE Transactions on Signal Processing, Vol. **53**, 2005, No 10, pp. 4020-4037.
29. T a n g, Z., S. W a n, X. Z h a n g. Lexicographical Framework for Image Hashing with Implementation Based on DCT and NMF. – Multimedia Tools and Applications, Vol. **52**, 2010, No 2-3, pp. 325-345.
30. Y u, Y., S. S u n. Image Robust Hashing Based on DCT Sign. – In: Proc. of 2006 International Conference Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP'06), 2006.
31. Q i n, C., C. C h a n g. Robust Image Hashing Using Non-Uniform Sampling in Discrete Fourier Domain. – Digital Signal Processing, Vol. **23**, 2013, No 2, pp. 578-585.
32. S w a m i n a t h a n, A. Robust and Secure Image Hashing. – IEEE Transactions on Information Forensics and Security, Vol. **1**, 2006, No 2, pp. 215-230.
33. S w a m i n a t h a n, A., Y. M a o, M. W u. Image Hashing Resilient to Geometric and Filtering Operations. – In: IEEE 6th Workshop on Multimedia Signal Processing, 2004, pp. 355-358.
34. A h m e d, F., M. Y. S i y a l, V. U. A b b a s. A Secure and Robust Hash-Based Scheme for Image Authentication. – Signal Processing, Vol. **90**, 2010, No 5, pp. 1456-1470.
35. L u, C. S., C. Y. H s u. Geometric Distortion Resilient Image Hashing Scheme and Its Applications on Copy Detection and Authentication. – Multimedia Systems, Vol. **11**, 2005, No 2, pp. 159-173.
36. L u, C. S., C. Y. H s u, S. W. S u n, P. C. C h a n g. Robust Mesh-Based Hashing for Copy Detection and Tracing of Images. – In: IEEE International Conference on Multimedia and Expo (ICME), 2004, pp. 731-734.
37. M e i x n e r, A., A. U h l. Robustness and Security of a Wavelet-Based CBIR Hashing Algorithm. – In: Proc. of 8th Workshop on Multimedia and Security, 2006, pp. 140-145.
38. M i c h a k, M. K., R. V e n k a t e s a n. New Iterative Geometric Methods for Robust Perceptual Image Hashing. – ACM Workshop: Security and Privacy on Digital Rights Management DRM, 2001, pp. 13-21.
39. M o n g a, V., B. L. E v a n s. Perceptual Image Hashing via Feature Points: Performance Evaluation and Tradeoffs. – IEEE Transactions on Image Processing, Vol. **15**, 2006, No 11, pp. 3453-3466.
40. V e n k a t e s a n, R., S. M. K o o n, M. H. J a k u b o w s k i, P. M o u l i n. Robust Image Hashing. – In: International Conference on Image Processing, 2000, pp. 664-666.
41. Y a n g, S. H., C. F. C h e n. Robust Image Hashing Based on SPIHT. – In: Proc. of IEEE International Conference Information Technology: Research & Education (IIH-MSP'06), 2006, pp. 110-114.
42. K a r s h, R. K., R. H. L a s k a r, A d i t i. Robust Image Hashing through DWT-SVD and Spectral Residual Method. – EURASIP Journal on Image and Video Processing, Vol. **2017**, 2017, No 31, pp. 1-17.
43. L e f e b v r e, F., J. C r y z, B. M a c q. A Robust Soft Hash Algorithm for Digital Image Signature. – In: Proc. of IEEE International Conference Image Processing, 2003, pp. 495-498.
44. L e f e b v r e, F., B. M a c q, J. D. L e g a t. RASH: RAdon Soft Hash Algorithm. – In: 11th European Signal Processing Conference, 2002, pp. 1-4.
45. M a l k i n, M. R. V e n k a t e s a n. The Randlet Transform: Applications to Universal Perceptual Hashing and Image Authentication. – In: Proc. of Allerton Conference, 2004.

46. S e o, J. S., J. H a i t s m a, T. K a l k e r, C. D. Y o o. A Robust Image Fingerprinting System Using the Radon Transform. – Signal Processing: Image Communication, Vol. **19**, 2006, No 4, pp. 325-339.
47. W u, D., X. Z h o u, X. N i u. A Novel Image Hash Algorithm Resistant to Print-Scan. – Signal Processing, Vol. **89**, 2009, No 12, pp. 2415-2424.
48. C h e n, C. C., S. L. H s i e h. Using Binarization and Hashing for Efficient SIFT Matching. – Journal of Visual Communication & Image Representation, Vol. **30**, 2015, pp. 86-93.
49. L i n g, H., L. Y a n, F. Z o u, C. L i u, H. F e n g. Fast Image Copy Detection Approach Based on Local Fingerprint Defined Visual Words. – Signal Processing, Vol. **93**, 2013, No 8, pp. 2328-2338.
50. L v, X., Z. J. W a n g. Perceptual Image Hashing Based on Shape Contexts and Local Feature Points. – IEEE Transactions on Information Forensics and Security, Vol. **7**, 2012, No 3, pp. 1081-1093.
51. L i, Y., Z. L u, C. Z h u, X. N i u. Robust Image Hashing Based on Random Gabor Filtering and Dithered Lattice Vector Quantization. – IEEE Transactions on Image Processing, Vol. **21**, 2012, No 4, pp. 1963-1980.
52. L i u, F., L. C h e n g, H. L e u n g, Q. F u. Wave Atom Transform Generated Strong Image Hashing Scheme. – Optical Communications, Vol. **285**, 2012, No 24, pp. 5008-5018.
53. L i u, Y. L., Y. X i a o. A Robust Image Hashing Algorithm Resistant against Geometrical Attacks. – Radio Engineering, Vol. **22**, 2013, No 4, pp. 1072-1081.
54. W a n g, L., X. J i a n g, D. H u, D. Y e , S. L i a n. Robust Perceptual Image Hash Using Gabor Filters. – In: International Conference on Multimedia Information Networking and Security, 2009, pp. 53-56.
55. Y a n g, B., F. G u, X. N i u. Block Mean Value Based Image Perceptual Hashing. – In: International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2006, pp. 167-172.
56. H e r n a n d e z, R. A. P., B. M. K u r k o s k i. Robust Image Hashing Using Image Normalization and SVD Decomposition. – In: IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS), 2011, pp. 1-4.
57. K o z a t, S. S., R. V e n k a t e s a n, M. K. M i c h a k. Robust Perceptual Image Hashing via Matrix Invariants. – IEEE International Conference on Image Processing (ICIP), 2004, pp. 3443-3446.
58. L v, X., Z. J. W a n g. An Extended Image Hashing Concept: Content-Based Fingerprinting Using FJLT. – EURASIP Journal on Information Security, 2009, pp. 1-17.
59. M o n g a, V., M. K. M i c h a k. Robust and Secure Image Hashing via Non-Negative Matrix Factorizations. – IEEE Transactions on Information Forensics and Security, Vol. **2**, 2007, No 3, pp. 376-390.
60. T a n g, Z., S. W a n g, X. Z h a n g, W. W e i, S. S u. Robust Image Hashing for Tamper Detection Using Non-Negative Matrix Factorization. – Journal of Ubiquitous Convergence and Technology, Vol. **2**, 2008, No 1, pp. 18-26.
61. K a r s h, R. K., R. H., L a s k a r, B. B. R i c h h a r i y a. Robust Image Hashing Using Ring Partition-PGNMF and Local Features. – SpringerPlus, Vol. **5**, 2016, No 1, pp. 1-20.
62. M o n g a, V., B. L. E v a n s. Robust Perceptual Image Hashing Using Feature Points. – In: IEEE International Conference on Image Processing, 2004, pp. 677-680.
63. R o y, S., Q. S u. Robust Hash for Detecting and Localizing Image Tampering. – In: IEEE International Conference on Image Processing, 2007, pp. 117-120.
64. T a n g, Z., S. W a n g, X. Z h a n g, W. W e i. Structural Feature-Based Image Hashing and Similarity Metric for Tampering Detection. – Fundamenta Informaticae, Vol. **106**, 2011, No 1, pp. 75-91.
65. X i a o f e n g, W., P. K e m u, Z. X i a o r u i, Z. W a n g, L. L u, X. J i a n r u. A Visual Model Based Perceptual Image Hash for Content Authentication. – IEEE Transactions on Information Forensics Security, Vol. **10**, 2015, No 7, pp. 1336-1349.

66. C h e n, Y., W. Y u, J. F e n g. Robust Image Hashing Using Invariants of Tchebichef Moments. – Optik, Vol. **125**, 2014, No 19, pp. 5582-5587.
67. Z h a o, Y., S. W a n g, X. Z h a n g, H. Y a o. Robust Hashing for Image Authentication Using Zernike Moments and Local Features. – IEEE Transactions on Information Forensics and Security, Vol. **8**, 2013, No 1, pp. 55-63.
68. Z h a o, Y., W. W e i. Perceptual Image Hash for Tampering Detection Using Zernike Moments. – In: Proc. of IEEE International Conference on Progress in Informatics and Computing, 2010, pp. 738-742.
69. T a n g, Z., L. H u a n g, Y. D a i, F. Y a n g. Robust Image Hashing Based on Multiple Histograms. – International Journal of Digital Content Technology and its Applications (JDCTA), Vol. **6**, 2013, No 23, pp. 39-47.
70. T a n g, Z., X. Z h a n g, L. H u a n g, Y. D a i. Robust Image Hashing Using Ring-Based Entropies. – Signal Processing, Vol. **93**, 2013, No 7, pp. 2061-2069.
71. T a n g, Z., X. Z h a n g, S. Z h a n g. Robust Perceptual Image Hashing Based on Ring Partition and NMF. – IEEE Transactions on Knowledge and Data Engineering, Vol. **26**, 2014, No 3, pp. 711-724.
72. G e r o l d, L., U. A n d r e a s. Key-Dependent JPEG2000 Based Robust Hashing for Secure Image Authentication. – EURASIP Journal on Information Technology, Vol. **8**, 2008, No 1, pp. 1-19.
73. H a d m i, A., W. P u e c h, B. A. E s S a i d, A. A. O u a h m a n. A Robust and Secure Perceptual Hashing System Based on Quantization Step Analysis. – Signal Processing: Image Communications, Vol. **28**, 2013, No 8, pp. 929-948.
74. J o h n s o n, M., K. R a m c h a n d r a n. Dither Based Secure Image Hashing Using Distributed Coding. – In: Proceedings of IEEE International Conference Image Processing, 2003, pp. 751-754.
75. K a n g, L., C. L u, C. H s u. Compressive Sensing Based Image Hashing. – In: Proc. of IEEE International Conference on Image Processing, 2009, pp. 1285-1288.
76. K h e l i f i, F., J. J i a n g. Perceptual Image Hashing Based on Virtual Watermark Detection. – IEEE Transactions on Image Processing, Vol. **19**, 2010, No 4, pp. 981-994.
77. L v, X., Z. J. W a n g. Compressed Binary Image Hashes Based on Semisupervised Spectral Embedding. – IEEE Transactions on Information Forensics and Security, Vol. **8**, 2013, No 11, pp. 1838-1849.
78. S k r e p t h, C. J., A. U h l. Robust Hash Functions for Visual Data: An Experiment Comparison. – Pattern Recognition and Image Analysis, Vol. **2652**, 2003, pp. 986-993.
79. S u n, R., X. Y a n, Z. D i n g. Robust Image Hashing Using Locally Linear Embedding. – In: International Conference on Computer Science and Service System, 2011, pp. 715-718.
80. X i a n g, S., H. J. K i m, J. H u a n g. Histogram-Based Image Hashing Scheme Robust against Geometric Deformations. – In: Proc. of 9th Workshop on Multimedia & Security, 2007, pp. 121-128.
81. Z o u, F., H. F e n g, H. L i n g, C. L i u, L. Y a n, P. L i, D. L i. Nonnegative Sparse Coding Induced Hashing for Image Copy Detection. – Neurocomputing, Vol. **105**, 2013, pp. 81-89.
82. T a n g, Z., L. H u a n g, X. Z h a n g, H. L a o. Robust Image Hashing Based on Color Vector Angles and Canny Operator. – International Journal of Electronics and Communications(AEU), Vol. **70**, 2016, No 6, pp. 833-841.
83. F r i d r i c h, J., M. G o l j a n. Robust Hash Functions for Digital Watermarking. – In: Proc. on IEEE International Conference on Information Technology: Coding and Computing, 2000, pp. 178-183.
84. G u o, X. C., D. H a t z i n a k o s. Content-Based Image Hashing via Wavelet and Radon Transform. – In: Advances in Multimedia Information Processing – PCM 2007, LNCS, 4810, 2007, pp. 755-764.

85. L a r a j i, I. H., L. G h o u t i, E. H. K h i a r i. Perceptual Hashing of Color Images Using Hyper Complex Representation. – In: Proc. of the IEEE International Conference on Image Processing (ICIP'13), 2013, pp. 4402-4406.
86. M i c h a k, M. K., R. V e n k a t e s a n. A Perceptual Audio Hashing Algorithm: A Tool for Robust Audio Identification and Information Hiding. – In: Proc. of 4th International Workshop on Information Hiding, Springer-Verlag, 2001, pp. 51-65.
87. M o n g a, V., D. V a t s, B. L E v a n s. Image Authentication under Geometric Attacks via Structure Matching. –In: IEEE International Conference Multimedia and Expo, 2005.
88. O u, Y., K. H. R h e e. A Key-Dependent Secure Image Hashing Scheme by Using Radon Transform. – 2009 International Symposium on Intelligent Signal Processing and Communications Systems (ISPACS), 2009, pp. 595-598.
89. R e y, C., J. L. D u g e l a y. A Survey of Watermarking Algorithms for Image Authentication. – EURASIP Journal on Applied Signal Processing, Vol. **1**, 2002, pp. 613-621.
90. S u n, Q., S. F. C h a n g. A Robust and Secure Media Signature Scheme for JPEG Images. – Journal of VLSI Signal Processing Systems for Signal, Image and Video Technology, Vol. **41**, 2005, No 3, pp. 305-317.
91. T a n g, Z., Y. D a i, X. Z h a n g, S. Z h a n g. Perceptual Image Hashing with Histogram of Color Vector Angles. – In: 8th International Conference on Active Media Technology(AMT'12), 2012, pp. 237-246.

Received: 01.11.2018; Second Version:10.03.2019; Accepted: 25.03.2019